

12  
ADA115584

ISI/SR-81-20

1980  
**Annual Technical Report**  
VOL. 2

December 1979 - September 1980

A Research Program in Computer Technology

Prepared for the  
**Defense Advanced Research Projects Agency**



DTIC FILE COPY



UNIVERSITY OF SOUTHERN CALIFORNIA

4676 Admiralty Way Marina del Rey California 90291

INFORMATION SCIENCES INSTITUTE

|                              |
|------------------------------|
| DISTRIBUTION STATEMENT A     |
| Approved for public release; |
| Distribution Unlimited       |

82 06 14 006

## Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE  |   | READ INSTRUCTIONS<br>BEFORE COMPLETING FORM   |
|--|---|---|
| 1. REPORT NUMBER<br>ISI/SR-81-20   | 2. GOVT ACCESSION NO.<br><i>AD-A225</i> | 3. RECIPIENT'S CATALOG NUMBER<br><i>584</i>   |
| 4. TITLE (and Subtitle)<br>1980 Annual Technical Report:<br>A Research Program in Computer Technology Vol. 2   |   | 5. TYPE OF REPORT & PERIOD COVERED<br>Annual Technical Report<br>December 1979 - September 1980 |
| 7. AUTHOR(s)<br>ISI Research Staff   |   | 6. PERFORMING ORG. REPORT NUMBER<br>MDA903 80 C 0523  |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>USC/Information Sciences Institute<br>4676 Admiralty Way<br>Marina del Rey, CA 90291  |   | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS<br>ARPA Order # 2223                |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Defense Advanced Research Projects Agency<br>1400 Wilson Blvd.<br>Arlington, VA 22209   |   | 12. REPORT DATE<br>August 1981  |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)<br>-----  |   | 13. NUMBER OF PAGES<br>24   |
| 16. DISTRIBUTION STATEMENT (of this Report)<br><br>This document is approved for public release and sale;<br>distribution is unlimited.  |   | 15. SECURITY CLASS. (of this report)<br>Unclassified  |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)<br><br>-----  |   | 18a. DECLASSIFICATION/DOWNGRADING SCHEDULE<br><br>-----   |
| 18. SUPPLEMENTARY NOTES<br><br>-----   |   |   |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number)<br>1. device fabrication, device testing, integrated-circuit-oriented language, MOS Implementation Service, silicon compilation, VLSI design, VLSI design library, wafer testing<br>2. computing resources, network security, network survivability, user training<br>3. address space limitations, C, Interlisp, LISP dialects, UNIX, VAX  |   |   |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number)<br><br>This report summarizes the research performed by USC/Information Sciences Institute from December 1, 1979, to September 30, 1980, for the Defense Advanced Research Projects Agency under contract MDA903 80 C 0523. The ISI program covered by this contract consisted of three research areas: VLSI - development of a low-cost, fast turnaround LSI/VLSI device fabrication service to support VLSI research, and research on the VLSI design problem; Strategic C3 System Experiment Support - development of a detailed technical plan for the creation of a survivable message system |   |   |

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 68 IS OBSOLETE  
S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. ABSTRACT (continued)

and data bases through multiple copies of the critical components and data across the ARPANET and provision of the core facilities necessary to implement the plan; Interlisp - development and maintenance of a portable, large address-space Interlisp implementation.

|                    |                         |
|--------------------|-------------------------|
| Accession For      |                         |
| NTIS GRA&I         |                         |
| DTIC TAB           |                         |
| Unannounced        |                         |
| Justification      |                         |
| By _____           |                         |
| Distribution/      |                         |
| Availability Codes |                         |
| Dist               | Avail and/or<br>Special |
| A                  |                         |



Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

**ISI/SR-81-20**

**1980  
Annual Technical Report  
VOL. 2**

**December 1979 - September 1980**

**A Research Program in Computer Technology**

Prepared for the  
**Defense Advanced Research Projects Agency**

**Effective date of contract: 1 October 1979**

**Contract expiration date: 30 September 1981**

**Principal Investigator and Executive Director:** **Keith W. Uncapher  
(213) 822-1511**

**Deputy Director:** **Thomas O. Ellis  
(213) 822-1511**

This research is supported by the Defense Advanced Research Projects Agency under Contract No. MDA903 80 C 0523 ARPA Order No. 2223.

Views and conclusions contained in this report are the authors' and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. Government or any other person or agency connected with them.

This document is approved for public release and sale; distribution is unlimited.



**UNIVERSITY OF SOUTHERN CALIFORNIA**

**4676 Admiralty Way Marina del Rey California 90291**

**INFORMATION SCIENCES INSTITUTE**



## CONTENTS

**Summary** *iv*

**Executive Overview** *v*

1. **VLSI** *1*
2. **Strategic C3 System Experiment Support** *13*
3. **Interlisp** *17*

## SUMMARY

This report summarizes the research performed by USC/Information Sciences Institute from December 1, 1979, to September 30, 1980, for the Defense Advanced Research Projects Agency under contract MDA903 80 C 0523.

The ISI program covered by this contract consisted of three research areas:

- *VLSI* - development of a low-cost, fast turnaround LSI/VLSI device fabrication service to support VLSI research, and research on the VLSI design problem;
- *Strategic C3 System Experiment Support* - development of a detailed technical plan for the creation of a survivable message system and data bases through multiple copies of the critical components and data across the ARPANET and provision of the core facilities necessary to implement the plan;
- *Interlisp* - development and maintenance of a portable, large address-space Interlisp implementation.

## EXECUTIVE OVERVIEW

The University of Southern California Information Sciences Institute (ISI) is a large information processing research center located in Marina del Rey, California.

ISI's principal focus is research in the field of information processing and digital communication. A majority of the research is application and systems oriented. ISI maintains a strong basic research program to support the application and systems focus. The Institute also is committed to a support role providing both general-purpose and special-purpose computing to a very large number of external users, as well as supplying most of ISI's internal needs.

The research programs at ISI sponsored by DARPA under contract MDA903 80 C 0523 are summarized below.

**VLSI.** This project is aimed at the development of a low-cost, fast turnaround LSI/VLSI device fabrication service to support a geographically distributed VLSI research community with no direct access to VLSI fabrication, but with access to a computer communication network. A special effort has been made to implement an LSI fabrication service having high-level automatic-access control and authentication and protection mechanisms. The motivation for this service is to provide an "almost interactive" LSI chip design environment with a design-to-packaged LSI device turnaround time of a few weeks. We believe this service will cause a revolution in system architecture research as dramatic as the introduction of interactive computation was to programming. In addition, we are also researching the design problem, from algorithms to "silicon compilation" through an integrated-circuit-oriented language.

**Strategic C3 System Experiment Support.** DARPA has defined an experiment in Strategic C3 systems to be conducted in cooperation with the WWMCCS System Engineer (WSE) and the Strategic Air Command (SAC). The concept of the experiment is to demonstrate and evaluate the use of new technologies (such as the ARPANET, packet radio, network security, and distributed knowledge-base techniques) for strategic command, control, and communication. The DARPA experiment is defined as having three phases: Phase I is planned to demonstrate air-to-surface packet radio links and gateways into the ARPANET as a first step in evaluating the feasibility of a truly survivable strategic network. Phase II is directed toward creating a survivable message system and data bases through multiple copies of the critical components and data across the ARPANET. Phase III will address the feasibility of rapid reconstitution of a strategic network by deployment of packet radio networks to reconnect surviving elements of the network. ISI is assisting in the development of a detailed technical plan for Phase II and providing the core facilities necessary to implement the plan.

**Interlisp.** This project is developing and will maintain a portable, large address-space Interlisp implementation. The first version is being done for the DEC VAX computer.

# 1. VLSI

## **Research Staff:**

Danny Cohen  
Yehuda Afek  
Ron Ayres  
Joel Goldberg  
Gideon Hess  
Dave Johannsen  
Lee Richardson  
Vance Tyree

## **Support Staff:**

Victor Brown  
Rolanda Shelby

## 1.1 INTRODUCTION

This project is aimed at the development of a low-cost, fast turnaround LSI/VLSI device fabrication service to support a geographically distributed VLSI research community with no direct access to VLSI fabrication, but with access to a computer communication network.

A special effort has been made to implement an LSI fabrication service having high-level automatic-access control and authentication and protection mechanisms. The motivation for this service is to provide an "almost interactive" LSI chip design environment with a design-to-packaged LSI device turnaround time of a few weeks. We believe this service will cause a revolution in system architecture research as dramatic as the introduction of interactive computation was to programming.

In addition to the above work, we are also researching the design problem, from algorithms to "silicon compilation" through an integrated-circuit-oriented language.

### 1.1.1 Problem Statement--Fabrication

"Fabrication" is the entire process of converting designs in machine readable form (Caltech Intermediate Form--CIF--here) into actual devices. It includes the grouping and placement of projects on dies and wafers, converting CIF into the target PG (Pattern Generation) format, mask/reticle generation, wafer fabrication, wafer fabrication quality testing (wafer probing), device separation (scribing or sawing), packaging, bonding, and distribution.

- The DARPA VLSI design community needs fabrication capabilities in order to investigate the design methodology and architecture appropriate to VLSI where gate-count will exceed  $10^{*}6$  gates per device.
- The above fabrication capabilities must be of sufficiently low cost to be affordable by the research community.
- The turnaround time must be fast enough to support semiinteractive research. Design and fabrication must be decoupled in order to achieve fabrication line interaction similar to device-independent programming.
- Many fabricators must be involved in order to assure the continuous availability of custom VLSI fabrication.
- Since the designers are not involved in the interface with the fabrication vendors, it is essential that a new testing methodology, which does not depend upon testing the circuits submitted by the designers, be used to evaluate the quality of the fabrication.

- Design rules and an intermediate form for expressing their patterns (CIF) are available now only for a limited subset of NMOS. Since the move to new technologies (e.g., CMOS) is forthcoming, these capabilities have to be supplemented in the near future.
- Similarly, the geometric processing (the conversion of CIF into PG format) is available only for a certain configuration (MEBES-1X). It is expected that conversion into more configurations will become necessary soon; otherwise the choice of masks, vendors, and fabrication lines is unnecessarily limited.
- Wafer probing and packaging are necessary to assure the utility of the fabrication operation.

### 1.1.2 Problem Statement--Design

- The current state of the art in design methodology is not fit for very large scale integrations, which involve devices with  $10^{**5}$  to  $10^{**6}$  gates.
- Even with today's silicon technology, current design tools always require many man-years to produce substantially new designs.
- Design tools, such as special integrated-circuit-oriented languages (ICL), are needed both at the lower levels of design and implementation, at the middle levels of device architecture, and at the higher levels where algorithms are converted into device architecture.

## 1.2 TECHNICAL APPROACH AND GOALS

The work in the VLSI project is divided into two areas: (1) fabrication services for the DARPA VLSI community, and (2) research in the area of high-level design methodology.

### 1.2.1 The Fabrication Service

The fabrication service area of the VLSI project was scoped to achieve the following:

- Develop, operate, and maintain a MOS Implementation Service (MOSIS) for use by the DARPA VLSI research community. The major components of the MOSIS system are:
  - Interaction with the designers,
  - Handling of their CIF files,
  - Fabrication of E-beam masks (via subcontract),
  - Wafer fabrication (via subcontract),
  - Wafer probing and data analysis,
  - Wafer sawing, die packaging, and bonding (via subcontract),
  - Chip distribution.
- Improve the design of the MOSIS system to expand its capabilities, especially in access control, budget control, and management information.
- Implement the MOSIS system to address those concerns that become important in the context of handling tens of thousands of designs, such as efficiency, impact on system resources, and the interaction with the file archive system.
- Interface to as many mask houses and fabrication lines as is practically possible, as part of promoting the "broker/foundry" concept.
- Coordinate the development of testing procedures required for quality control of the MOSIS operation and keep databases for use at any time in yield data analysis.

- Control the performance of both testing and packaging as required for the fabrication service offered by MOSIS.
- Design cell libraries to support absolute geometry entries, provide access control to the library, and automatically distribute the information about the content of the library and its usage.
- Maintain the design library and provide the necessary support for it.

### 1.2.2 Research

Research began in the following areas:

- Investigating further development of ICL, an integrated-circuit-oriented programming language.
- Investigating the development and application of high-level "silicon compilation."
- Studying the methodology of converting algorithms into VLSI-oriented architectures.

## 1.3 SUMMARY OF PROGRESS

### 1.3.1 MOSIS--the DARPA fabrication broker

After a period of debugging, the MOSIS system became operational in January 1981. The MOSIS system accepts VLSI designs expressed in CIF [1] and handles all the fabrication steps needed in order to provide the designers with actual chips. The major components of the MOSIS system are:

- Interaction with the designers,
- Handling of their CIF files,
- Converting CIF into PG tapes,
- Fabrication of E-beam masks (by subcontract to external mask houses),
- Wafer fabrication (also subcontracted to external vendors),
- Wafer probing and data analysis,
- Wafer sawing, die packaging, and bonding (also with external subcontracts),
- Chip distribution.

As part of running this service for the DARPA community, the MOSIS crew interacts with various designers in order to help them use the system, receive important information, etc. The crew also interacts with the various vendors in order to establish many of the parameters required for each run. As part of its mission, the MOSIS crew is continually seeking new vendors and encouraging them to offer custom VLSI fabrication service.

A major activity of MOSIS is the development of a testing methodology required for efficient decoupling of the design from the fabrication process. These activities are described in detail below.

MOSIS serves the DARPA VLSI research community including (among others) Caltech, Massachusetts Institute of Technology, Carnegie-Mellon University, UC Berkeley, Lincoln Laboratory, ISI, Jet Propulsion Lab, and the National Bureau of Standards. The emphasis of the MOSIS operation is to support VLSI implementation, independent of the specific details (idiosyncrasies) of those fabrication lines that MOSIS happens to use, by providing support of the NMOS design rules as

described in [2]. However, MOSIS is capable of supporting other technologies (e.g., CMOS-SOS), even though these technologies are not yet offered to the general user community.

### 1.3.2 The User Interface

The MOSIS user interface is based entirely on an exchange of ARPANET messages between the user and MOSIS. Unless a user specifically asks to communicate with a person (usually only to handle a special request), the entire message exchange is handled automatically by MOSIS. This user interface works so smoothly that several users were under the impression that the interchange was handled manually by the MOSIS crew.

User interaction with the system has the following steps:

1. Approval of the user privilege to use MOSIS services,
2. Information requests,
3. Definition of new projects,
4. Submission of CIF for fabrication,
5. Status and progress reports from MOSIS,
6. Distribution of fabricated devices (chips).

Steps 2 through 5 above are all handled by messages according to formats specified in the MOSIS User Manual, which is available online. Users unfamiliar with MOSIS can "bootstrap" themselves to this information by following the instructions sent to them by MOSIS. There is always a response to messages even if they are not in the proper MOSIS format.

User messages to MOSIS consist of REQUESTs that have several parameters.

MOSIS uses several mechanisms to protect the propriety of design information. These include passwords both for the individual designers and for each project, so that users who cooperate in working on certain projects do not have to provide each other access to all of their other designs.

### 1.3.3 The Fabrication Interface

The MOSIS interface to the fabrication facilities has several aspects. The most important interaction is with the wafer fabrication facilities. Depending on the various specifications of the particular fabricator, MOSIS may have to adjust several parameters, such as geometric compensation (bloats and shrinks), and generate the appropriate set of masks.

MOSIS shields the designers from many details involved in the fabrication process. We are very proud that most MOSIS users do not even know about these details and that they need not know about them in order to obtain fabricated IC's. All of these details are handled by the "control protocol" of the MOSIS interface to the fabrication facilities.

The bulk of the interface, "data protocol," is a set of pattern generation (PG) tapes. These are used by a mask house to produce a set of glass masks (by using an E-beam pattern generator), which are required for fabrication. The next portion of the fabrication interface generates a set of bonding maps (automatically produced by MrBill, the MOSIS geometry processor) needed to turn wafers into usable devices.

### 1.3.4 MrBill, the Geometry Processor

All IC-specific processing of the MOSIS service is done by an ICL program called MrBill. MrBill accepts individual projects submitted by people over the ARPANET, through MOSIS. Each design is represented in CIF.

MrBill reads in CIF files, checks their validity, and converts them into an internal representation. MrBill then "packs" the projects onto individual dies (chips) in such a manner that no project interferes with any other project. It is this packing that allows the costs of fabrication to be distributed among many people, rendering a small cost per project and a more efficient use of silicon.

MrBill allows the operator to assign names to the newly formed chips. These names appear directly on the chips in aluminum. This assures a permanent naming convention so that our users can identify their projects to us in terms we readily understand.

Finally, MrBill translates each chip into MEBES format. It is this format that the mask fabricator's E-beam mask machine understands. The geometric processing is complete at this point.

MrBill's database also produces bonding diagrams for the people who package our chips. A bonding diagram shows the bonders how to connect the chip to the pins on the package.

### 1.3.5 MOSIS Quality Assurance

In order to evaluate the quality of the MOSIS fabrication runs, it is necessary to have a set of test vehicles that can be placed on each wafer or die. In the MOSIS runs, test vehicles serve as a common ground between designers and fabricators. A designer executes his design on the assumption that some basic elements (such as transistors and interconnects) behave in a particular way and on the expectation that a large number of elements can be defectless. To verify his design, he submits it to a MOSIS run. Before investing energy into verifying his design by testing the operability of a fabricated circuit, he should be assured that these assumptions regarding the behavior of the basic elements are correct. Measurements on test vehicles either on the same die or on the same wafer containing the designer's circuit are the only practical way to provide such assurances. Such test vehicles also benefit the fabricator. The quality of his product can be more accurately and fairly judged on the operability of standard test circuits rather than the operability of an unproven and perhaps badly designed circuit submitted by a designer.

Such test vehicles are also of considerable value to the broker bringing designers and fabricators together. It allows that broker to make meaningful assessments of the capabilities of various fabricators.

#### Test Vehicle Developments for MOSIS

Test vehicles for MOSIS runs have taken two forms: (1) test strips on each die containing a relatively small number of structures, each of which is tested with automated equipment prior to separation of the wafer into dies, and (2) a test die included in each fabrication run that contains large test structures used to evaluate the fabrication yield for the design rules supported by MOSIS.

### Test Strip

The test strip currently in use contains a few basic transistors and interconnects that allow measurement of some basic parameters at each die location on each wafer from a fabrication run. The test strip measurements are taken by automatic wafer-probing and provide information about transistor threshold voltage, sheet resistance of various layers, inverter gain (for inverters of various ratios), contact resistance, leakage current, and inverter threshold.

In addition to the basic transistor measurements, a few basic building blocks are used to ascertain circuit performance. Examples of these building blocks include: an inverter; an inverter fed with a minimum size pass transistor; long conductors for direct measurement of timing delays; and a few stages of a dynamic shift register to measure dynamic node storage times.

Data collected from the test strip are processed to gather statistical information about the fabrication run. These processed data are printed out in various formats to show test results by wafer ID and by die location on each wafer. The MOSIS crew uses these summarized results in the following ways:

1. Wafer acceptance: The test strip results are used to determine if the fabricator has met required transistor thresholds, inverter thresholds, inverter gain, etc., for each wafer lot.
2. Wafer selection: Each fabrication run yields a few more wafers than are required to satisfy the needs of the users. Wafers to be cut, packaged, wire bonded, and shipped to the users are selected by using test strip results to find the "best of the lot."
3. SPICE parameters: Basic parameters for use in SPICE simulation of circuits (implemented with the design rules supported by MOSIS) are extracted from test strip measurements.

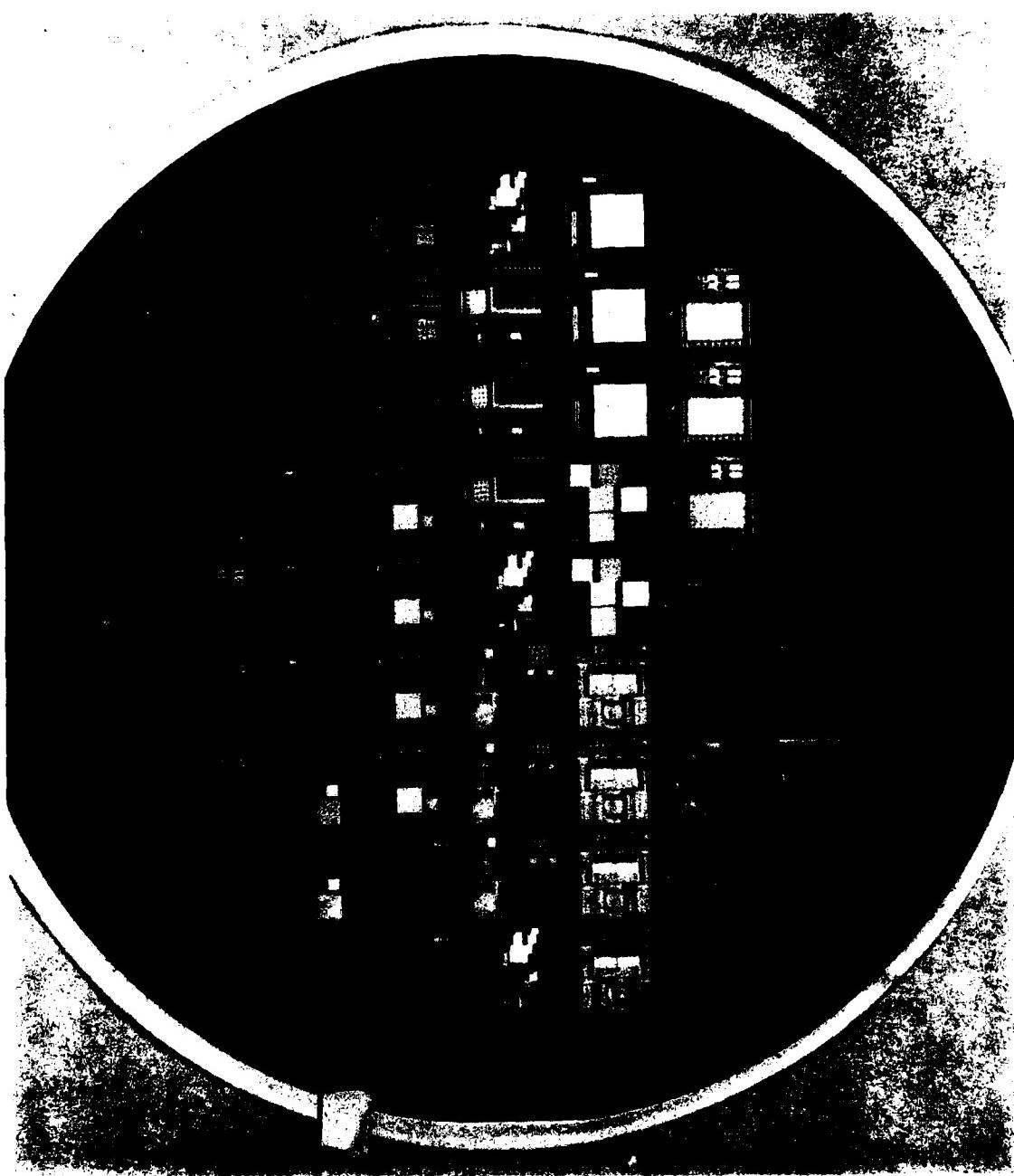
### Test Die

The test die is intended to provide users with information about yield for each fabrication run. Measurements of structures on this die provide a designer with information about the probability of his having a defectless die. One of the structures we have been testing is a large (approximately 4K bits) random access memory based on lambda minimum spacing design rules, which will give us a direct measure of fatal defects per unit area.

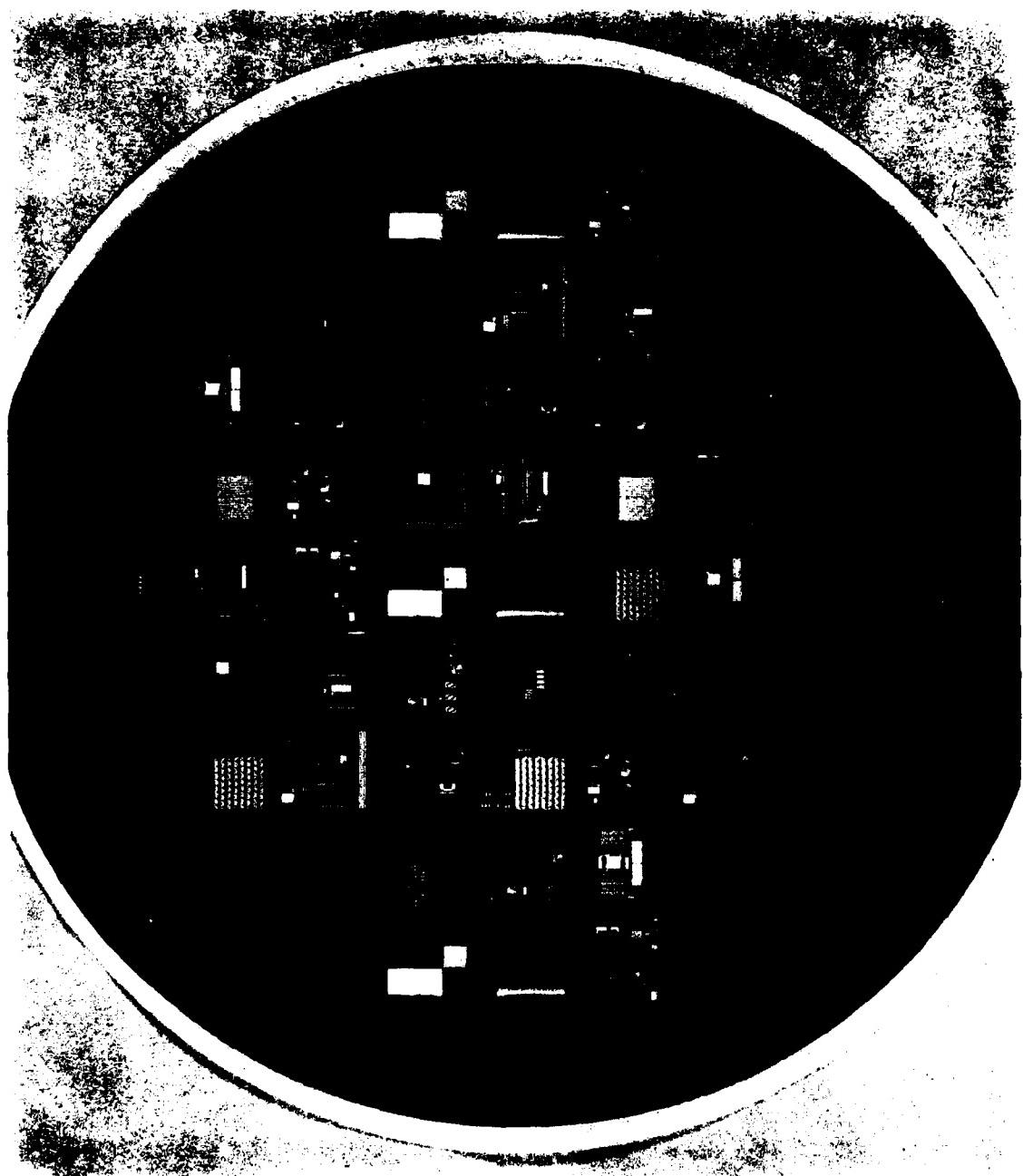
Other methods for obtaining fabrication yield are being explored.

### Fabrication Runs Completed

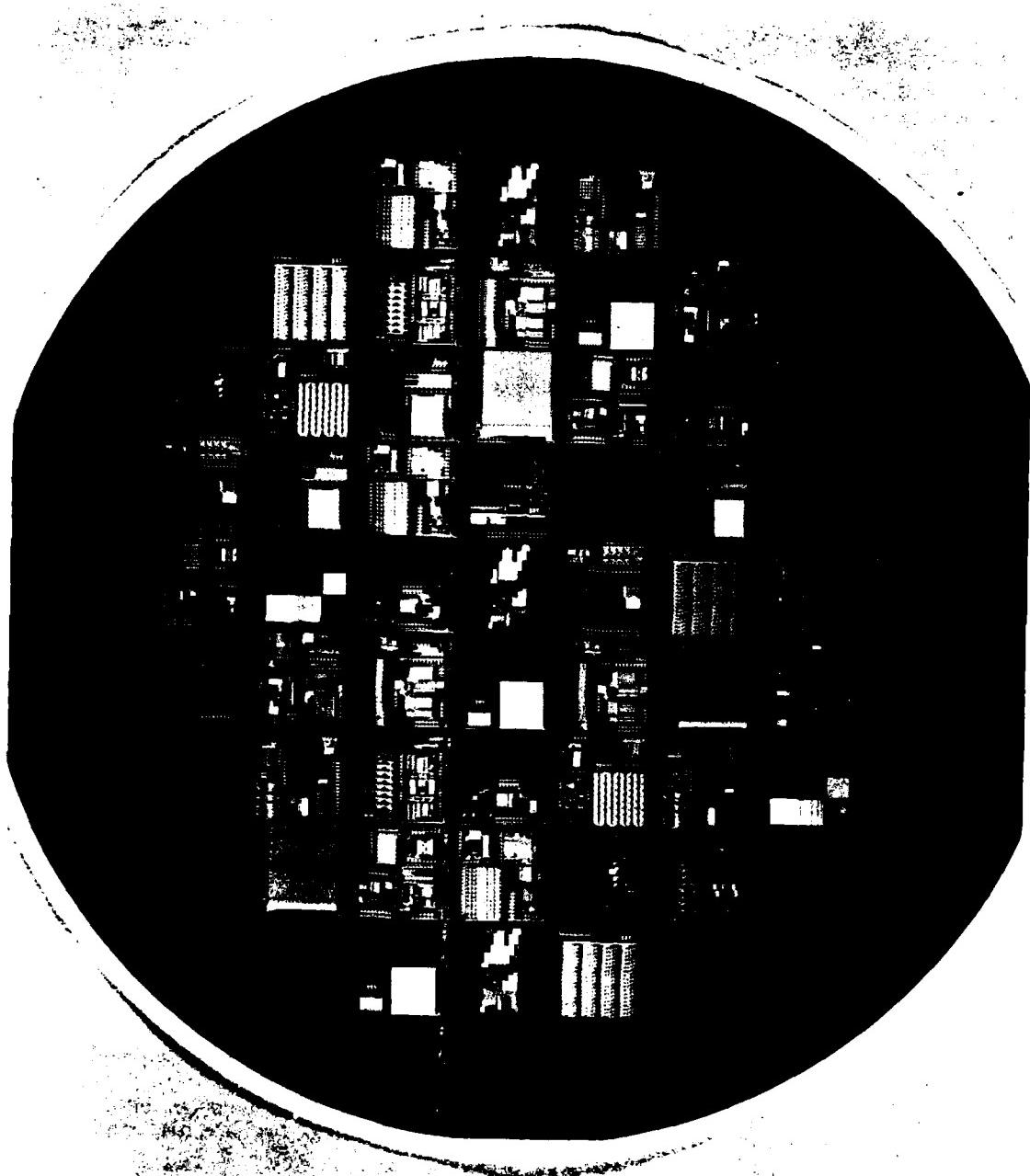
In August 1980 MOSIS accepted designs for the first fabrication run using the software developed for automatic interaction with users. This run had 65 projects submitted by designers from 8 organizations: ISI, UCLA, Caltech, Jet Propulsion Lab, Stanford University, National Bureau of Standards, Carnegie-Mellon University, and Washington University at St. Louis. These projects were packed into 18 dies on the wafer. Since this run was considered a trial for the MOSIS service, users were informed of a small possibility that some unforeseen problems might result in a less than optimum yield. The run turned out to be good. Reports from users were positive. A photograph of this trial run wafer (M08B) is included (Fig. 1) to illustrate the placement of the projects on the wafer. Other recent fabrication runs (M11E and M11F) are also pictured (Figs. 2 and 3).



**Figure 1-1: Photograph of trial run wafer M08B**



**Figure 1-2: Photograph of fabrication run M11E**



**Figure 1-3: Photograph of fabrication run M11F**

### 1.3.6 MOSIS User Service--The VLSI Design Library

The MOSIS VLSI Design Library currently offers some basic circuits for bonding pads, super-buffers, shift registers, and PLA fragments. These circuits were acquired from Xerox Palo Alto Research Center in the form of CIF files. The circuits had been used in IC's during the early Multiproject Chip fabrication runs and were fully tested. Before these circuits were released for general use, the CIF files were examined to extract geometric interface coordinates included in separate circuit documentation files distributed to users upon request. The basic circuits currently in the library are designed both for lambda = 2.5 microns and for lambda = 2.0 microns. New circuits are being acquired and will be added to the library as they are tested and documented.

### 1.3.7 Research

The research effort at ISI has been proceeding in the area of silicon compilation and VLSI-based computer architecture.

#### Silicon Compilation

We are pursuing a novel and promising way to produce integrated circuit designs as easily as software programs are produced today. Where the presently accepted methods for designing chips involve human interaction with every possible detail, we are pursuing the automatic generation of designs from clear and simple behavioral descriptions.

The production of software would come to a screeching halt if programmers used tools like the ones used presently in IC designs. Today ICs are typically designed with an interactive graphics editor. (It is interesting to note that interactive graphics has been applied even to software development, e.g., the GRAIL project.)

Software people know that most of the boring and error-prone work is done for them by software compilers, e.g., FORTRAN, BASIC, Pascal, or LISP. The programmer no longer specifies bits, or even machine instructions, but specifies his intent more directly. This is always done in two general steps:

1. Someone proposes and implements a new programming language.
2. Programmers specify their programs in this new language.

There are two advantages to the introduction of a new language:

1. The language provides convenient, concise, and clear notations for many popular and useful fragments of computation.
2. The implementor of the language can provide each of these notations very efficiently, once and for all. Programmers need not reinvent each technique over and over again.

Finally, since a language is finite in its implementation, but infinite in the number and variety of program specifications that it can implement, the language implementors can economically spend enormous amounts of time, effort, and thought producing very efficient implementations.

Silicon compilers not only provide an implementation in silicon, but also introduce new silicon languages. The hard part of creating a silicon compiler is the production of a language that simultaneously satisfies the following:

1. The language is convenient to use, i.e., it is close to the natural domain of the human designer.
2. The language is also sufficiently understood and restrictive so that a reliable implementation can be created to produce relatively efficient silicon designs.

#### Present Silicon Compiler Status

We are examining two existing silicon compiler prototypes, RELAY and Bristle-Blocks. RELAY tends to be general purpose, whereas Bristle Blocks tends to be very efficient in the silicon result. Bristle Blocks takes advantage of design techniques developed at Caltech that are especially well suited for the production of microprocessors. RELAY, in contrast, accepts the specification for and implements any synchronous digital system. RELAY readily produces hierarchies of state machines.

Upon examination of these two contrasting silicon compilers, we are noting certain regularities, invariants, and simplifications that can serve in the design of more flexible and more efficient silicon compilers. We have also noted that techniques from both RELAY and Bristle Blocks are needed for attaining very practical, wide-application silicon compilers.

At ISI we have fabricated and tested a chip produced by RELAY, and the chip works. We are now working to test chips produced by Bristle Blocks. (The chips produced by Bristle Blocks are more complex than the test case chosen for RELAY, a simple 4-bit counter implemented in a hierarchy that results in a two-by-two array of single-bit counters).

#### 1.4 FUTURE WORK

During the next reporting period the VLSI Project will extend the MOSIS service to include the following:

1. Smaller feature sizes. The current NMOS design rules can be supported at a feature size of 5 microns. It is expected that this will be reduced to 4 microns soon.
2. Additional technologies. Currently the MOSIS service only supports NMOS technology fabrication. Work is currently under way to expand the number of technologies available to MOSIS users. The next technology that is likely to be supported is CMOS.
3. A smoother, more interactive user interface. The interface is planned for the latter part of the next fiscal year.
4. Improved and expanded testing capabilities. Wafer probe testing is performed upon every wafer fabricated for the user community. A standard test structure on each project die is used to measure the quality of the fabrication. This testing is being improved and expanded to provide information on average yield for typical fabrication runs and to extract and distribute transistor model parameters to the users. An effort is under way to develop yield measuring test structures that will accurately model the expected yield for typical user designs. In addition, the current test strip and drop-in test die are being redesigned to simplify testing and improve the accuracy of parameter extraction.

## REFERENCES

1. Hon, R. W., and C. H. Sequin, *A Guide to LSI Implementation* (second edition), Xerox Palo Alto Research Center, Technical Report SSL-79-7, January 1980.
2. Mead, C. A., and L. A. Conway, *Introduction to VLSI Systems*, Addison-Wesley, Reading, Mass., 1980.

## 2. STRATEGIC C3 SYSTEM EXPERIMENT SUPPORT

**Research Staff:**  
Louis Gallenson  
Rob Stotz  
Charlie Brown  
Chloe Holg

**Support Staff:**  
Joyce Reynolds  
Lisa Vail

### 2.1 INTRODUCTION

DARPA has defined an experiment in Strategic C3 systems to be conducted in cooperation with the WWMCCS System Engineer (WSE) and the Strategic Air Command (SAC). The concept of the experiment is to demonstrate and evaluate the use of new technologies (such as the ARPANET, packet radio, network security, and distributed knowledge-base techniques) for strategic command, control, and communication. Specific goals are to:

- Demonstrate and evaluate the survivability of multinode computer-communication networks, including the ARPANET and packet radio, especially for remote access from both airborne platforms and surface sites.
- Explore replication and reconstitution of critical knowledge bases on this network in the face of a loss of a large number of links.
- Demonstrate and evaluate the rapid reconstitution of a network by rapid deployment of packet radio nets to reestablish connectivity between surviving elements of the network.
- Conduct experiments and exercises to evaluate the utility of such a network on a distributed knowledge base to support postattack C3 activities.

The DARPA experiment is defined as having three phases:

1. Phase I is planned to demonstrate air-to-surface packet radio links and gateways into the ARPANET as a first step in evaluating the feasibility of a truly survivable strategic network.
2. Phase II is directed toward creating a survivable message system and data bases through multiple copies of the critical components and data across the ARPANET.
3. Phase III will address the feasibility of rapid reconstitution of a strategic network by deployment of packet radio networks to reconnect surviving elements of the network.

### 2.2 PROBLEM BEING SOLVED

ISI is assisting in the development of a detailed technical plan for Phase II and providing the core facilities necessary to implement the plan. Specifically, SAC must have fairly broad-based experience with ARPANET-based on-line interactive computing prior to the introduction of Phase II. Installation of modems, installation of up to 30 interactive CRT terminals, user training, system software training and support, and on-site maintenance of equipment were the focus for FY'80.

## 2.3 GOALS AND APPROACH

The major goals of this reporting period were to:

- Retrofit 30 Military Message Experiment (MME) HP-2649 terminals for SAC requirements.
- Install the required terminal communication system at SAC-Offutt Field, including modems, concentrators, and local lines where required.
- Install and debug 30 terminals and associated printers at SAC Headquarters.
- Establish connectivity from all terminals and printers to the ARPANET via the GWC-TIP (Air Force Global Weather Central-Terminal Interface Processor).
- Initiate user training in message systems, text editors, and administrative support functions at ISI for the persons selected by SAC to be trainers (training the trainers, for on-site training at SAC).
- Establish computing resources at ISI for use by SAC in administrative support functions and in launching SAC system programming efforts under TOPS-20 on the ISI computers.
- Hire and train a full-time maintenance liaison to work at SAC-Offutt.
- Monitor via the ARPANET the progress of SAC system software development, the progress of SAC trainers, and any systemwide problems the SAC users may experience. These tasks were begun with ISI's cooperation.

## 2.4 PROGRESS

We converted 28 HP-2649 MME terminals to standard HP2645 alphanumeric CRT terminals and shipped them to SAC-Offutt. Three units will serve as spares; twenty-five have been installed and are fully operational. Nineteen additional terminals were procured and installed at SAC-Offutt: five HP-2639 printing terminals, five Xerox 1750 high-quality printers, two HP-2649G graphics terminals, five TI Silent 700 portable terminals, and two HP plotters. We designed a local communication system at SAC, linking terminals to concentrators to the TIP in order to accommodate the geographical distribution of terminals throughout the base and the extremely noisy local lines at SAC. Modems were acquired and installed where long lines required retransmission of the RS232 interface signals.

ISI helped to prepare SAC personnel for the experiment by means of both training at ISI and periodic visits from R. Stotz and C. Holg. SAC system programmers were trained at ISI and are being given further assistance via the ARPANET. We are also providing half a man-year of system software development at ISI in support of SAC requirements. A full-time ISI hardware maintenance liaison (C. Brown) has been trained at ISI and assigned to SAC-Offutt.

Because of ARPANET loading problems complicated by GWC's novel use of the net, ISI has spent considerable time measuring the host computer and associated net connection to determine choke points in the subnet (with the cooperation of Bolt Beranek and Newman), the net-host interface problems, and the unique tape drive associated with the GWC-TIP. Also, because of extremely heavy loads on all ISI TOPS-20 systems, SAC's response-time problems have been extreme. The problem appears to be similar for the user communities at Ft. Bragg, Ga., and the AFS at Gunter, Ala. Measurements and experiments have helped to reduce the congestion (see Volume 1, pp. 131-132, "Netloading"), and the SAC community has benefited. The problems persist as a result of the large concentration of users vying for limited ARPANET resources. We are continuing to work with IPTO and BBN to minimize response-time problems and improve service to SAC, Ft. Bragg, and Gunter.

The GWC-TIP supports a tape drive, unique for standard TIPs on the ARPANET, which consumes all of the TIP resources when reading or writing tapes. This causes periodic breakdown of communications between terminals at SAC and the host at ISI. The problem will be solved in the near future when the GWC-TIP no longer supports the tape drive. Additional performance improvements are expected when a second, more powerful TIP is installed at SAC-Offutt.

## 2.5 IMPACT

As a result of extensive instrumentation efforts at ISI and BBN, the subnet congestion problem has been reduced considerably. We have offloaded ISIE to the point that SAC users are assured a load average of five or less, which assures good host response time.

User training has produced proficient users, but the number will remain small until the serious delays at SAC are eliminated.

## 2.6 FUTURE WORK

Although the major thrust of the DARPA Strategic C3 system experiment is well defined, the detailed definition of the Phase II program is still evolving. ISI will continue to assist DARPA in the shaping of this program, working to satisfy the communication and computer resource requirements of SAC headquarters. In particular, ISI will do the following:

- Continue to provide on-site maintenance support for the required equipment.
- Continue to plan and assist in implementing improved SAC connectivity to the ARPANET.
- Install and maintain terminals and communication equipment for connection of two numbered Air Force bases to the ARPANET for their participation in the experiment.
- Continue to supply programming and engineering support for Phase II of the experiment.

### 3. INTERLISP

**Research Staff:**  
Robert Balzer  
Dave Dyer  
Hans Koomen  
Bill Rizzi

**Support Staff:**  
Joan Elliott  
Lisa Vail

#### 3.1 PROBLEM BEING SOLVED

This project is establishing an ARPA support center for Interlisp in order to provide a large address-space version for the VAX, and is designing a portable version for migration to new, more cost-effective machines as they become commercially available.

#### 3.2 BACKGROUND

A serious problem has arisen within the research community as the research projects using LISP have developed larger and more complex programs. A basic limitation of the PDP-10, on which current systems are implemented, has been reached. The address space available (256K words) is too small; current research projects routinely require several times as much space. A recent ARPA-sponsored workshop on symbolic computation estimated that minimum requirements are 8 million addressable words and that these requirements are expected to grow towards 2130 words; conventional wisdom estimates this growth at .5\*bits/year.

Experience has consistently shown that programming costs rise exponentially as resource limitations are approached. Thus, many LISP-based research projects are expending considerable resources in system programming efforts to mitigate the effects of this address-space limitation.

This difficulty provided the incentive for the community to consider various alternatives. The first question was whether LISP was the best language for such large, complex research projects. The result of this discussion was the conclusion that there is no existing widely acceptable alternative (although this is an important research area). The second question was which dialect of LISP is best. There are two main dialects, Interlisp and Maclisp, which differ mainly in their control structures and the environment developed around the language. The user communities expressed strong preference for their own dialects. The Maclisp community is pursuing a number of alternatives, including two new dialects: CONS, implemented on new hardware (a specially designed LISP machine), and NIL, being implemented on an S1.

ISI's Interlisp project addresses the needs of the Interlisp community. For this community, the LISP machine option was rejected because of concern for the availability and maintainability of the hardware and because of dialect incompatibilities. Instead, the VAX computer was chosen as the initial machine for Interlisp implementation on the basis of its expected widespread use throughout the academic community and the opportunity for technology transfer such widespread use affords.

PRECEDING PAGE BLANK AND SEALED

It was further argued that no one machine would (or should) dominate Interlisp (or in fact IPTO) usage in the future as the PDP-10 has until now. Instead, new, more cost-effective machines will be appearing with increasing frequency. Therefore the new, large-address-space Interlisp should be constructed to be portable, and the Interlisp support center should migrate this system to new hardware on a periodic basis.

### 3.3 PROGRESS

This project started in April 1980, and in the intervening nine months we have evaluated several different implementation approaches. Inner virtual machine definitions (such as Byte LISP or PDP-11 LISP) were rejected because they inherently rely on extensive microcode support (beyond current VAX microcode capabilities). Extension of existing VAX LISP implementations was rejected because one (FRANZ LISP) represents only a small kernel LISP (also, proprietary issues concerning its compiler exist), while the other (NIL) is intended primarily for implementation on the S1.

That left as the best approach a direct implementation of Interlisp as defined in Xerox PARC's "Interlisp Virtual Machine" document. Initially, to speed development and reduce risk, we decided not to use VAX microcode and to suppress portability as a major design issue. During the tuning phase of the project, we will reconsider use of VAX microcode.

The VAX Interlisp will run under UNIX, the standard ARPA operating system for the VAX, and will be implemented in C, the standard implementation language for all UNIX systems. The use of these off-the-shelf components is expected to enhance the portability of the finished product. For example, the Nu environment at MII and the Spice environment at CMU are also based on UNIX-like operating systems and the C language. A compatible C compiler for TOPS-20 will also be available soon, allowing the possibility of porting our implementation back to the PDP-10, using the extended addressing mode available under TOPS-20 version 4.

We evaluated an existing implementation of the Interlisp Virtual Machine, called Multilisp, which was implemented in Pascal. We found it to be well done and well documented. With IPTO's participation, we negotiated a license that allows us to take this proprietary system, transliterate it into C, modify it so that it runs under UNIX, and (most significantly) extend it so that it supports the entire Interlisp environment. The resulting system will be nonproprietary and in the public domain.

We have converted most of the Multilisp system into C and are about to begin the checkout of the conversion. A VAX/780 dedicated to the project will be installed in January 1981.

A considerable amount of effort has been expended on identifying the machine and operating-system dependencies in the Interlisp environment; these must be addressed so that the environment itself can be ported. We expect this task to be a major part of the remaining effort.

### 3.4 MILITARY IMPACT

The military relevance of this effort arises from military use of the many ARPA-sponsored research projects based on Interlisp.

This project is designed to fulfill the VAX requirements and to lay the foundation for the portability requirement.

### 3.5 CURRENT WORK

The design of the Virtual Machine (VM) definition has been completed. It will be composed of some converted modules of Multilisp, some modules and parts of modules from Xerox's Interlisp-D (Byte LISP) and Interlisp-10, and some completely new pieces. There will be a very small amount of VAX assembly language.

We have produced a list of the LISP-coded modules that can be used "as is" or with limited modifications. A VAX code cross-compiler, operating in the current Interlisp-10 environment, has been written and partially debugged. Work on this will resume as soon as the VM can support it. During the initial stage of integration of LISP code into the developing Interlisp-VAX, code compiled on TOPS-20 will be transferred to the VAX for execution. VAX/TOPS-20 transfers will be supported by a limited FTP to be developed for use with our Ethernet. The PNG-11 will be called upon to support these transfers as well as TOPS-20/Alto transfers. The development of this hardware and software will be done by other ISI staff.

We expect the implementation of the VM to be completed by June 1981. The modifications of existing Interlisp-D and Interlisp-10 modules should be complete (but untested) at about the same time, and then the task of piling on the layers of the Interlisp environment will begin. We expect this task to be completed by January 1982. We are planning the initial release (of a tested, reliable product) in March 1982, with subsequent maintenance releases as necessary thereafter. During the period April to October 1982, the focus will be on completing the implementation documentation, providing user-service, maintaining the April release, and making some significant performance enhancements--especially applicable to very large programs. A third and final release is planned for October 1982.

The final release terminates the development aspects of the project. The required level of user-support and of maintenance should also decrease about this time (as the users become better acquainted with the system, and the rate of detecting bugs declines). Therefore, the staff will be reduced after October 1982.